

Claims

[system for]

- [including]
providing a first cache memory containing data;

providing a second cache memory containing

- the first cache memory is the same as at least some of the data of the second cache memory; and
 [cache selection hardware for selection]
 in response to a request for data that is stored in both the first cache memory and the
 second cache memory, choosing which one of the cache memories to use to obtain the data
 according to an access balancing technique.

2. A method, according to claim 1, wherein the access balancing technique includes at least one of: selection using round robin and selection based on statistics.

3. A method, according to claim 2, wherein selection based on statistics includes:

monitoring a number of accesses of the first cache memory over a predetermined amount of time;

monitoring a number of accesses of the second cache memory over the predetermined amount of time; and

selecting one of the cache memories having a lowest number of accesses over the predetermined amount of time.

4. A method, according to claim 1, further comprising:

subdividing the data into control data and into corresponding disk data.

5. A method, according to claim 4, wherein a first balancing technique is used to balance access to the control data and a second balancing technique is used to balance access to the disk data.

6. A method, according to claim 5, wherein the first and second balancing techniques are the same.

7. A method, according to claim 6, wherein the first and second access balancing techniques include at least one of: selection using round robin and selection based on statistics.

8. A method, according to claim 5, wherein the first and second balancing techniques are different.

9. A method, according to claim 8, wherein the first access balancing technique includes at least one of: selection using round robin and selection based on statistics.

10. A method, according to claim 9, further comprising:

in response to choosing the first cache memory for accessing the control data, accessing the second cache memory for the corresponding disk data; and

in response to choosing the second cache memory for accessing the control data, accessing the first cache memory for the corresponding disk data.

11. A method, according to claim 9, wherein the second access balancing technique includes at least one of: selection using round robin and selection based on statistics.

12. A method, according to claim 1, wherein choosing which one of the cache memories includes using specialized hardware.

13. The method of Claim 12, wherein the specialized hardware includes at least one chip.

5 14. The method of Claim 12, further comprising:

receiving, by the specialized hardware, requests from for at least one of the first and second cache memories.

15. The method of Claim 14, further comprising:

processing, by the specialized hardware, the requests in accordance with the access balancing technique.

16. The method of Claim 15, further comprising:

subdividing the data into control data and corresponding disk data.

17. The method of Claim 16, further comprising:

using the specialized hardware to access at least one of said control data and said corresponding disk data.

20 18. A system for managing data in a cache comprising:

a first cache memory including data;

a second cache memory including data, wherein at least some of the data included in the first cache memory is the same as at least some of the data of the second cache memory; and

cache selection hardware for selecting, in response to a request for data that is stored in both the first cache memory and the second cache memory, which one of the first and second cache memories to use to obtain the data in accordance with an access balancing technique.

19. The system of Claim 18, wherein said access balancing technique is a first access balancing technique and the system further comprises:

machine executable code for selecting, in response to a request for data that is stored in both the first cache memory and the second cache memory, which one of the first and second cache memories to use to obtain the data in accordance with a second access balancing technique.

20. The system of Claim 19, wherein the data includes at least one of control data and disk data.

21. The system of Claim 20, wherein said cache selection hardware is used in implementing one of the first and second access balancing techniques, and the machine executable code is used in implementing a different one of the balancing techniques than the cache selection hardware.

22. The system of Claim 20, wherein said cache selection hardware is used in implementing at least one of said first and second balancing techniques, and the machine executable code is used in implementing at least one of said first and second balancing techniques.

23. The system of Claim 21, wherein said first and second access balancing techniques includes at least one: selection using round robin and selection based on statistics.

24. The system of Claim 22, wherein said first and second access balancing techniques includes at least one: selection using round robin and selection based on statistics.

5 25. The system of Claim 24, wherein the cache selection hardware further includes:

first hardware for selecting the first cache memory for accessing the control data; and
second hardware, coupled to the first hardware, for accessing the second cache memory for the disk data corresponding to the control data when said first cache memory is used for accessing the control data.

10 26. The system of Claim 25, wherein the cache selection hardware further includes:

third hardware for selecting the second cache memory for accessing the control data; and
fourth hardware, coupled to the third hardware, for accessing the first cache memory for the disk data corresponding to the control data when said second cache memory is used for accessing the control data.

15 27. The system of Claim 24, wherein said machine executable code further includes machine executable code for:

selecting the first cache memory for accessing the control data; and
for accessing the second cache memory for the disk data corresponding to the control data when said first cache memory is used for accessing the control data.

28. The system of Claim 27, wherein said machine executable code further includes machine executable code for:

selecting the second cache memory for accessing the control data; and

accessing the first cache memory for the disk data corresponding to the control data

when said second cache memory is used for accessing the control data.

29. The system of Claim 18, further comprising:

a first bus used in sending communications between at least one of: the first cache memory and the second cache memory, and the cache selection hardware; and

a second bus used in sending communications between at least one of: the first cache memory and the second cache memory, and the cache selection hardware.

30. The system of Claim 27, wherein said first bus is used in facilitating communications between said cache selection hardware and said first cache memory, and said second bus is used in facilitating communications between said cache selection hardware and said second cache memory.

31. A ^{method} computer program product ^{of} for managing data in a cache, comprising:

machine executable code for providing a first cache memory containing data;

machine executable code for providing a second cache memory containing data, wherein

at least some of the data of the first cache memory is the same as at least some of the data of the second cache memory; and

[machine executable code for, in response to a request for data that is stored in both the first cache memory and the second cache memory, choosing which one of the cache memories to use to obtain the data according to an access balancing technique.

5 32. The computer program product, according to Claim 31, wherein the [machine executable code for choosing which one of the cache memories to use to obtain the data according to an access balancing technique, further includes machine executable code for at least one of: selection using round robin and selection based on statistics.

3 33. The computer program product, according to claim 32, wherein the [machine executable code for choosing which one of the cache memories to use to obtain the data according to an access balancing technique, further includes machine executable code for:

monitoring a number of accesses of the first cache memory over a predetermined amount of time;

monitoring a number of accesses of the second cache memory over the predetermined amount of time; and

selecting one of the cache memories having a lowest number of accesses over the predetermined amount of time.

7 20 34. The computer program product, according to claim 31, further comprising:

[machine executable code for subdividing the data into control data and into corresponding disk data.

35. The computer program product, according to claim 34, further comprising machine executable code for a first balancing technique used to balance access to the control data and a second balancing technique is used to balance access to the disk data.

6

5 36. The computer program product, according to claim 35, wherein the first and second balancing techniques are the same.

7

37. The computer program product, according to claim 36, wherein the machine executable code for the first and second access balancing techniques include machine executable code for at least one of: selection using round robin and selection based on statistics.

8

38. The computer program product, according to claim 35, wherein the first and second balancing techniques are different.

9

39. The computer program product, according to claim 38, wherein the machine executable code for the first access balancing technique includes machine executable code for at least one of: selection using round robin and selection based on statistics.

10

40. The computer program product, according to claim 9, further comprising machine executable code for:

in response to choosing the first cache memory for accessing the control data, accessing the second cache memory for the corresponding disk data; and

